REMARKS

The enclosed is responsive to the Examiner's Notice of Non-Compliant Amendment Office Action mailed on January 29, 2008. In the office action, the Examiner noticed claim 142, on page 7, did not begin on a new line. Applicants have corrected claim 142 to begin on a new line pursuant to the Examiner's suggestion. At the time the examiner mailed the Office Action claims 126-167 were pending. By way of the present response Applicant has: 1) amended claims 126, 130, 131, 132, 137, 141, and 143; 2) added no new claims; and 3) canceled claims 155-167. As such, claims 126-154 are now pending. The Applicant respectfully request reconsideration of the present application and the allowance of all claims now represented.

Examiner Interview

Applicant would like to thank Examiner Bashore for graciously conducting an interview on Oct. 26, 2007 to discuss the claims of the application and the significance of the prior art in relation to the claims. In response to suggestions made by the examiner during the course of the interview regarding the clarity of what was previously claimed, Applicant has amended some of the claims as indicated above.

Claim Rejections

35 U.S.C. 103(a) Rejections

The Examiner rejected claims 126-127, 129-134, 136-138, 140-146, 148-154 under 35 U.S.C. 103(a) as being unpatentable over Omniform User's Manual (hereinafter "OmniForm"), Caere Corporation, released March 22, 1999 (as evidenced cited PR NewsWire article), pages 1-108, 173-199, in view of Hitchcock, et al., U.S. Patent 6,460,042 (hereinafter "Hitchcock").

OmniForm describes the generation of an HTML form and the editing of an existing form by a user. (OmniForm, e.g., pages 25-43.) This editing may be performed through a GUI. (OmniForm, e.g., page 32.)

Hitchcock, at least in the sections cited by the office action, describes generating a form from directives. (Hitchcock, col.10 ll.42-65.)

The combination does not describe what Applicant claims require. Specifically, with respect to claim 126, the combination does not describe:

> A method comprising:
> receiving, through a network, a form authored using a form authoring language, the form containing one or more input fields;
> parsing the received form to identify the input fields contained in the received form;
> providing a graphical user interface to allow identification of one or more actions to be associated with the identified input fields upon subsequent specific submission of a specific instance of the form by a third party, the provided graphical user interface being dependent on the identified input fields;
> automatically generating a program code to carry out the actions associated with the identified input fields, wherein no modifications to the input fields of the form are made by the generation of the program code.

Both of the cited references describe the generation of a form. Neither describes the generation of program code for identified input fields of an existing form. Accordingly, the combination does not describe what Applicant claim 126 requires. Claims 127-131 are dependent on claim 126 and are allowable for at least the same reason.

With respect to claim 132, the combination does not describe:

> A server comprising:
> a communications device connected to a network to receive a form from a client connected to the network, the form authored using a form authoring language and containing one or more input fields;
> a memory coupled to the communications device to store the received form;
> a parser module coupled to the memory to parse the received form to identify the input fields contained in the received form;

a configurer module coupled to the parser module to create a graphical user interface based on the input fields identified by the parser module, and to provide the graphical user interface using the communications device to allow the identification of actions to be associated with the identified input fields upon subsequent submission of a specific instance of the form by a third party;

a code generator module coupled to the configurer module to automatically generate a program code to carry out the actions identified, wherein no modifications to the input fields of the form are made by the identification of actions to be associated with the input fields.

Both of the cited references describe the generation of a form. Neither describes the generation of program code for identified input fields of an existing form. Accordingly, the combination does not describe what Applicant claim 132 requires. Claims 133-136 are dependent on claim 132 and are allowable for at least the same reason.

With respect to claim 137, the combination does not describe:

A machine-readable medium containing data representing instructions that, when executed by a processor, cause the processor to perform operations comprising:

receiving, through a network, a form authored using a form authoring language, the form containing one or more input fields;

parsing the received form to identify the input fields contained in the received form;

providing a graphical user interface to allow identification of one or more actions to be associated with the identified input fields upon subsequent submission of a specific instance of the form by a third party, the provided user graphical user interface being dependent on the identified input fields;

automatically generating a program code to carry out the actions associated with the identified

input fields, wherein no modifications to the input fields of the form are made by the generation of the program code.

Both of the cited references describe the generation of a form. Neither describes the generation of program code for identified input fields of an existing form. Accordingly, the combination does not describe what Applicant claim 137 requires. Claims 138-142 are dependent on claim 137 and are allowable for at least the same reason.

With respect to claim 143, the combination does not describe:

A method comprising:

receiving, through a network, a form authored using a form authoring language, the form containing one or more input fields;

parsing the received form to identify the input fields contained in the received form;

providing a graphical user interface to allow identification of an action[[s]] to be associated with one of the identified input fields upon subsequent submission of a specific instance of the form by a third party, the provided user graphical user interface being dependent on the identified input fields;

automatically generating a program code to carry out the actions associated with the identified input field, wherein the program code does not modify the input fields of the form;

receiving a specific instance of the form, including data input into the input fields; and

executing the program code on the received data from the specific submission of the form to carry out the actions associated with the identified input field.

Both of the cited references describe the generation of a form. Neither describes the generation of program code for identified input fields of an existing form. Accordingly, the combination does not describe what Applicant claim 143

requires. Claims 144-154 are dependent on claim 143 and are allowable for at least the same reason.

In light of the comments above, Applicants respectfully requests the allowance of all claims.

CONCLUSION

Applicants respectfully submit that all rejections have been overcome and that all pending claims are in condition for allowance.

If there are any additional charges, please charge them to our Deposit Account Number 02-2666. If a telephone conference would facilitate the prosecution of this application, the Examiner is invited to contact Edwin H. Taylor at (408) 720-8300.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: 2/26/08 /Ryan W. Elliott/

Ryan W. Elliott
Reg. No.: 60,156

1279 Oakmead Parkway
Sunnyvale, CA 94085-4040
(408) 720-8300